Amendments to the Claims

      The following Listing of Claims will replace all prior versions and listings of claims in the application.

Listing of Claims

1. (Currently amended) A method for handling plug-and-play events occurring at a client, said method comprising ~~the steps of~~:

      (a) providing a client communicating with a server over a network using a presentation-level protocol;

      (b) detecting a plug-and-play event notification regarding a device in communication with the client;

      (c) redirecting said event notification to the server from the client~~;~~; and

      (d) receiving, <u>in response to the redirection of the event notification, a command</u> from the server, <u>the</u> ~~a~~ command directed to said device.

2. (Currently amended) The method of claim 1 wherein <u>redirecting said event notification</u> ~~step (c)~~ further comprises ~~the steps of~~:

      (c-1) generating a context identifier, said context identifier representing a virtual COM port;

      (c-2) binding the context identifier to the event notification; and

      (c-3) transmitting the bound context identifier and event notification to the server.

3. (Currently amended) The method of claim 1 wherein <u>redirecting said event notification includes redirecting said event notification via</u> ~~step (c) uses~~ a virtual channel ~~to redirect said event notification~~.

4. (Currently amended) The method of claim 1, wherein <u>receiving a command from the server</u> ~~step (d)~~ further comprises ~~the steps of~~:

      (d-1) receiving from a server a command including a generated context identifier;

      (d-2) identifying the device using the context identifier; and

(d-3) issuing a command to the identified device.

5. (Original) The method of claim 1 wherein said event notification is generated as a result of a device arrival.

6. (Original) The method of claim 5 wherein said command is an open command.

7. (Original) The method of claim 1 wherein said event notification is generated as a result of a device removal.

8. (Original) The method of claim 7 wherein said command is a close command.

9. (Original) The method of claim 1 wherein said event notification is associated with at least one of a GUID, vendor ID, product ID and actual device name.

10. (Original) The method of claim 1 wherein the device in communication with the client uses one of the USB (Universal Serial Bus) protocol, IEEE 1394 protocol, Bluetooth protocol, wi-fi protocol, wireless protocol, and infrared (IR) protocol to communicate with the client.

11. (Currently amended) A method for handling plug-and-play events occurring at a client in communication with a server using a presentation-level protocol, said method comprising ~~the steps of~~:

    a) receiving from said client a plug-and-play event notification regarding a device in communication with the client;

    b) notifying an application program hosted by the server of the occurrence of the event notification;

    c) receiving<u>, in response to notification of the occurrence of the event notification, a command</u> from the application program hosted by the server<u>, the</u> ~~a~~ command directed to the device; and

    d) transmitting to the client a command directed to the device.

12. (Currently amended) The method of claim 11 wherein the event notification ~~received in step (a)~~ from the client is received over a virtual channel.

13. (Currently amended) The method of claim 11 wherein the event notification ~~received in step (a)~~ includes a context identifier bound to the event notification, said context identifier representing a virtual COM port.

14. (Currently amended) The method of claim 11, further comprising ~~the steps of~~: creating a server-unique name to identify the device connected to the client that generated the event notification, said server unique name used in mapping the client device to a specific session on the server established by the presentation level protocol.

15. (Currently amended) The method of claim 11 wherein <u>notifying an application program</u> ~~step (b)~~ further comprises ~~the step of~~: transmitting the event notification to applications communicating with the server within the session.

16. (Currently amended) The method of claim 11 wherein <u>notifying an application program</u> ~~step (b)~~ further comprises ~~the step of~~: transmitting the event notification only to applications communicating with the server which have previously registered a callback for a type of event causing the event notification.

17. (Original) The method of claim 11 wherein said event notification is generated as a result of a device arrival.

18. (Original) The method of claim 17 wherein said command is an open command.

19. (Original) The method of claim 11 wherein said event notification is generated as a result of a device removal.

20. (Original) The method of claim 19 wherein said command is a close command.

21. (Cancelled)

22. (Cancelled)

23. (Currently amended) A method for handling events occurring at a client in communication with a server using a presentation-level protocol, said method comprising ~~the steps of~~:

a) receiving from said client an event notification regarding a device in communication with the client;

b) notifying an application program hosted by the server of the occurrence of the event notification;

c) receiving, in response to notification of the occurrence of the event notification, a command from the application program hosted by the server, the ~~a~~ command directed to the device; and

d) transmitting to the client a command directed to the device.

24. (Currently amended) A method for informing a server about the presence of devices connected to a client, said method comprising ~~the steps of~~:

(a) providing a client communicating with a server over a network using a presentation-level protocol;

(b) emulating a plug-and-play event notification regarding a device in communication with the client;

(c) redirecting said emulated event notification to the server over a network; and

(d) receiving, in response to the redirection of the event notification, a command from the server, the ~~a~~ command directed to said device.

25. (Currently amended) The method of claim 24 wherein redirecting said emulated event ~~step (c)~~ further comprises ~~the steps of~~:

(c-1) generating a context identifier, said context identifier representing a virtual COM port;

(c-2) binding the context identifier to the emulated event notification; and

(c-3) transmitting the bound context identifier and emulated event notification to the server.

26. (Currently amended) The method of claim 24 wherein the redirection of the emulated event notification ~~in step (c)~~ uses a virtual channel.

27. (Currently amended) The method of claim 24, wherein <u>receiving the command from the server</u> ~~step (d)~~ further comprises ~~the steps of~~:

(d-1) receiving from a server a command identifying the generated context ID;

(d-2) identifying the device using the context; and

(d-3) issuing a command to the identified device.

28. (Currently amended) The method of claim 27 wherein the emulated event notification received ~~in step (b)~~ from the client is received over a virtual channel.

29. (Currently amended) The method of claim 27 wherein the emulated event notification received ~~in step (b)~~ includes a context ID bound to the emulated event notification.

30. (Currently amended) The method of claim 27 wherein <u>redirecting said emulated event</u> ~~step (c)~~ further comprises ~~the step of~~: broadcasting the emulated event notification to applications communicating with the server.

31. (Currently amended) The method of claim 27 wherein <u>redirecting said emulated event</u> ~~step (c)~~ further comprises ~~the step of~~: transmitting the emulated event notification only to applications communicating with the server which have previously registered a callback with the server for a type of event causing the emulated event notification.

32. (Currently amended) The method of claim 24, wherein said client is a proxy client on a server <u>interfacing</u> ~~, said server interfaced~~ with at least one additional server.

33. (Currently amended) A method for informing a server about the presence of network resources connected to a proxy client, said method comprising ~~the steps of~~:

a) emulating a plug-and-play event notification regarding a network resource in communication with the proxy client;

b) redirecting said emulated event notification to a server; and

c) receiving, <u>in response to the redirection of the event notification, a command</u> from the server, <u>the</u> ~~a~~ command directed to said network resource.

34. (Currently amended) ~~An article of manufacture having embodied thereon~~ <u>A</u> computer-readable program <u>having executable instructions</u> ~~means~~ for handling plug-and-play events occurring at a client communicating with a server over a network using a presentation-level protocol, the <u>computer readable medium</u> ~~article of manufacture~~ comprising:

~~computer-readable program means~~ <u>instructions</u> for detecting a plug-and-play event notification regarding a device in communication with the client;

~~computer-readable program means~~ <u>instructions</u> for redirecting said event notification to the server; and

~~computer-readable program means~~ <u>instructions</u> for receiving, <u>in response to the redirection of the event notification, a command</u> from the server, <u>the</u> ~~a~~ command directed to said device.

35. (Currently amended) The ~~article of manufacture of claim 34 wherein the~~ computer-readable <u>medium of claim 34</u> ~~program means for redirecting said event notification~~ further comprising ~~comprises~~:

~~computer-readable program means~~ <u>instructions</u> for generating a context identifier, said context identifier representing a virtual COM port;

~~computer-readable program means~~ <u>instructions</u> for binding the context identifier to the event notification; and

~~computer-readable program means~~ <u>instructions</u> for transmitting the bound context identifier and event notification to the server.

36. (Currently amended) The ~~article of manufacture~~ <u>computer readable medium</u> of claim 34

wherein instructions ~~the computer-readable program means~~ for redirecting said event notification include instructions that use ~~uses~~ a virtual channel to redirect said event notification.

37. (Currently amended) The computer readable medium ~~article of manufacture~~ of claim 34 wherein instructions ~~the computer-readable program means~~ for receiving a command, ~~from the server, a command directed to said device~~, further comprise~~comprises~~:

instructions ~~computer-readable program means~~ for receiving a command from a server that includes ~~a command including~~ the generated context identifier;

instructions ~~computer-readable program means~~ for identifying the device using the context identifier; and

instructions ~~computer-readable program means~~ for issuing a command to the identified device.

38. (Currently amended) The computer readable medium of ~~article of manufacture~~ of claim 34 wherein said event notification is generated as a result of a device arrival.

39. (Currently amended) The computer readable medium ~~article of manufacture~~ of claim 38 wherein said command is an open command.

40. (Currently amended) The computer readable medium ~~article of manufacture~~ of claim 34 wherein said event notification is generated as a result of a device removal.

41. (Currently amended) The computer readable medium ~~article of manufacture~~ of claim 40 wherein said command is a close command.

42. (Currently amended) The ~~article of manufacture~~ computer readable medium of claim 34 wherein said event notification is associated with at least one of a GUID, vendor ID, product ID and actual device name.

43. (Currently amended) The ~~article of manufacture~~ computer readable medium of claim 34 wherein the device in communication with the client uses one of the USB (Universal Serial Bus)

protocol, IEEE 1394 protocol, Bluetooth protocol, wi-fi protocol, wireless protocol, and infrared (IR) protocol to communicate with the client.

44. (Currently amended) <u>A computer readable medium having executable instructions</u> ~~An article of manufacture having embodied thereon computer-readable program means~~ for handling plug-and-play events occurring at a client communicating with a server over a network using a presentation-level protocol, <u>the computer readable medium</u> comprising:

 <u>instructions</u> ~~computer-readable program means~~ for receiving from said client a plug-and-play event notification regarding a device in communication with the client;

 <u>instructions</u> ~~computer-readable program means~~ for notifying an application program hosted by the server of the occurrence of the event notification;

 <u>instructions</u> ~~computer-readable program means~~ for receiving<u>, in response to notification of the occurrence of the event notification, a command</u> from the application program hosted by the server<u>, the</u> ~~a~~ command directed to the device; and

 <u>instructions</u> ~~computer-readable program means~~ for transmitting to the client a command directed to the device.

45. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 44 wherein the event notification is received over a virtual channel.

46. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 44 wherein the received event notification includes a context identifier bound to the event notification, said context identifier representing a virtual COM port.

47. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 44, further comprising:

 ~~computer-readable program means~~ <u>instructions</u> for creating a server-unique name to identify the device connected to the client that generated the event notification, said server unique name used in mapping the client device to a specific session on the server established by the presentation level protocol.

48. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 44 wherein the <u>instructions</u> ~~computer-readable program means~~ for notifying an application program hosted by the server of the occurrence of the event notification further <u>comprise</u>~~comprises~~: <u>instructions</u> ~~computer-readable program means~~ for transmitting the event notification to applications communicating with the server within the session.

49. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 44 wherein the <u>instructions</u> ~~computer-readable program means~~ for notifying an application program hosted by the server of the occurrence of the event notification further ~~comprises~~<u>comprise</u>: ~~computer-readable program means~~ <u>instructions</u> for transmitting the event notification only to applications communicating with the server which have previously registered a callback for a type of event causing the event notification.

50. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 44 wherein said event notification is generated as a result of a device arrival.

51. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 50 wherein said command is an open command.

52. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 44 wherein said event notification is generated as a result of a device removal.

53. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 52 wherein said command is a close command.

54. (Cancelled)

55. (Cancelled)

56. (Currently amended) <u>A computer readable medium</u> ~~An article of manufacture~~ having <u>executable instructions</u> ~~embodied thereon computer-readable program means~~ for handling events

occurring at a client communicating with a server over a network using a presentation-level protocol, the computer readable medium comprising:

computer-readable program means instructions for receiving from the client an event notification regarding a device in communication with the client;

computer-readable program means instructions for notifying an application program hosted by the server of the occurrence of the event notification;

computer-readable program means instructions for receiving, in response to notification of the occurrence of the event notification, a command from the application program hosted by the server, the a command directed to the device; and

computer-readable program means instructions for transmitting to the client a command directed to the device.

57. (Currently amended) A computer readable medium An article of manufacture having executable instructions embodied thereon computer-readable program means for informing a server about the presence of devices connected to a client communicating with a server over a network using a presentation-level protocol, the computer readable medium comprising:

computer-readable program means instructions for emulating a plug-and-play event notification regarding a device in communication with the client;

computer-readable program means instructions for redirecting said emulated event notification to a server; and

computer-readable program means instructions for receiving, in response to the redirection of the event notification, a command from the server, the a command directed to said device.

58. (Currently amended) The computer readable medium article of manufacture of claim 57 wherein the instructions computer-readable program means for redirecting said emulated event notification further comprise comprises:

computer-readable program means instructions for generating a context identifier, said context identifier representing a virtual COM port; computer-readable program means for binding the context identifier to the emulated event notification; and

~~computer-readable program means~~ <u>instructions</u> for transmitting the bound context identifier and emulated event notification to the server.

59. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 57 wherein the redirection of the emulated event notification uses a virtual channel.

60. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 57, wherein the <u>instructions</u> ~~computer-readable program means~~ for transmitting the bound context identifier and emulated event notification to the server further <u>comprise</u>~~comprises~~:

     <u>instructions</u> ~~computer-readable program means~~ for receiving from a server a command identifying the generated context ID;

     <u>instructions</u> ~~computer-readable program means~~ for identifying the device using the context; and

     ~~computer-readable program means~~ <u>instructions</u> for issuing a command to the identified device.

61. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 60 wherein the emulated event notification is received over a virtual channel.

62. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 60 wherein the emulated event notification includes a context ID bound to the emulated event notification.

63. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 60 wherein the <u>instructions</u> ~~computer-readable program means~~ for identifying the device using the context further <u>comprise</u>~~comprises~~: ~~computer-readable program means~~ <u>instructions</u> for broadcasting the emulated event notification to applications communicating with the server.

64. (Currently amended) The <u>computer readable medium</u> ~~article of manufacture~~ of claim 60 wherein the ~~computer-readable program means~~ <u>instructions</u> for identifying the device using the context further <u>comprise</u>~~comprises~~:

~~computer-readable program means~~ _instructions_ for transmitting the emulated event notification only to applications communicating with the server which have previously registered a callback with the server for a type of event causing the emulated event notification.

65. (Currently amended) The ~~article of manufacture~~ _computer readable medium_ of claim 57, wherein said client is a proxy client on a server, said server interfaced with at least one additional server.

66. (Currently amended) _A computer readable medium_ ~~An article of manufacture~~ having _executable instructions_ ~~embodied thereon computer-readable program means~~ for informing a server about the presence of network resources connected to a proxy client, _the computer readable medium_ comprising:

instructions ~~computer-readable program means~~ for emulating a plug-and-play event notification regarding a network resource in communication with the proxy client;

~~computer-readable program means~~ _instructions_ for redirecting said emulated event notification to a server; and

~~computer-readable program means~~ _instructions_ for receiving, _in response to the redirection of the event notification, a command_ from the server, _the_ ~~a~~ command directed to said network resource.

67. (Currently amended) A method for enumerating devices communicating with a client that have been mapped into a session on a server, said method comprising ~~the steps of~~:

launching an application in a user session on a server intercepting device enumeration methods in the server-based user session;

redirecting the device enumeration methods to the server;

emulating an arrival event for at least one device enumerated by the redirected method, said device being a device in communication with a client system that was mapped into the user session prior to said application launch; ~~and~~

notifying said application hosted by the server of the occurrence of the _arrival_ event notification~~.~~_; and_

<u>receiving, in response to notification of the occurrence of the arrival event, a command from the application hosted by the server, the command directed to the at least one device.</u>

68. (Currently amended) A method for handling plug-and-play events occurring at a client, said method comprising ~~the steps of~~:

     (a) detecting a plug-and-play event notification regarding a device communicating with the client via a USB connection on the client;

     (b) redirecting said event notification to a server over a network; and

     (c) receiving, <u>in response to the redirection of the event notification, a command</u> from the server, <u>the</u> ~~a~~ command directed to said device.

69. (Original) The method of claim 68 wherein said event notification is generated as a result of a device arrival.

70. (Original) The method of claim 69 wherein said command is an open command.

71. (Original) The method of claim 68 wherein said event notification is generated as a result of a device removal.

72. (Original) The method of claim 71 wherein said command is a close command.

73. (Original) The method of claim 68 wherein said event notification is associated with at least one of a GUID, vendor ID, product ID and actual device name.

74. (Currently amended) The method of claim 68 wherein <u>redirecting said event notification</u> ~~step (b)~~ further comprises ~~the steps of~~:

     (b-1) generating a context identifier, said context identifier representing a virtual COM port;

     (b-2) binding the context identifier to the event notification; and

     (b-3) transmitting the bound context identifier and event notification to the server.

75. (Currently amended) The method of claim 68 wherein <u>redirecting said event notification</u> ~~step (b)~~ <u>further comprises redirecting said event notification via</u> ~~uses~~ a virtual channel ~~to redirect said event notification~~.

76. (Currently amended) The method of claim 68, <u>receiving a command from the server</u> ~~wherein step (c)~~ further comprises ~~the steps of~~:

(c-1) receiving from a server a command including a generated context identifier;

(c-2) identifying the device using the context identifier; and

(c-3) issuing a command to the identified device.